What Is Claimed Is:

1. A method for automatically mapping state elements between a first circuit and a second circuit, comprising:

a) comparing, in a structural phase, structural features of state elements in the first circuit to structural features of state elements in the second circuit for equivalence;

b) determining, during the structural phase, mappings between state elements of the first circuit and the second circuit based on the comparison of the structural features;

c) accounting for don't care input conditions before comparing state element function;

d) detecting, in an inversion detection phase, the polarity of the mappings;

e) comparing, in a functional phase, the functionality of state elements in the first circuit to state elements in the second circuit for equivalence using a three-valued simulation;

f) determining further mappings based upon the functional comparison during the functional phase; and

g) detecting whether a threshold condition for completion of the mapping process is satisfied.

2. The method of claim 1, further comprising:

h) building, in the structural phase, initial fanin and fanout partial signatures for each state element in the first and second circuits;

i) determining whether at least one of a fanin partial signature and a fanout partial signature for a first unmapped state element in the first circuit is equivalent to a respective one of a fanin partial signature and a fanout partial signature of a second unmapped state element of the second circuit;

j) further determining if the first state element and the second state element are the only state elements which share an equivalent at least one of a fanin partial signature and fanout partial signature; and

k) mapping the first and second state elements to one another if it is determined that they are the only state elements which share the at least one of the fanin partial signature and fanout partial signature.

3. The method of claim 2, further comprising:

l) updating the fanin and fanout partial signatures of the state elements of the first and second circuits based on the mapping between the first and second state elements.

4. The method of claim 3, further comprising:

repeating steps i) through l) until a fix point is reached ending the structural phase, the fix point occurring at a completion of a number of repeated iterations during which no further mappings are determined.

5. The method of claim 2, wherein the fanin partial signature is a list of mapped inputs to the state element, and the fanout partial signature is an alphabetically sorted list of mapped outputs from the state element.

6. The method of claim 1, further comprising:

m) inputting random values, excluding don't care conditions, to each element of a pair of mapped state elements during the inversion detection phase, the random values being bit values for mapped inputs having a known polarity and being a third value for unmapped inputs and mapped inputs having an unknown polarity;

n) comparing output values from each element of the pair of state elements based on the input random values; and

o) determining whether the pair is one of direct and inverse mapped based on the comparison.

7. The method of claim 6, wherein the mapped inputs to each element of the pair of elements having a known polarity are one of direct mapped inputs and inverse mapped inputs, direct mapped inputs to each element being provided with equivalent random values, inverse mapped inputs to each element being provided with complementary random values.

8. The method of claim 7, further comprising:

p) updating the fanin and fanout partial signatures of the state elements of the

first and second circuits to reflect the determination of polarity for the pair of state elements.

9. The method of claim 6, further comprising:

q) validating the mapping of the pair of state elements if the comparison indicates that the mapping is one of a direct mapping and an inverse mapping.

10. The method of claim 6, further comprising:

simulating parallel versions of a pair of mapped state elements;

making parallel comparisons of the parallel versions; and

determining whether the pair is one of direct and inverse mapped based on the comparisons of the parallel versions.

11. The method of claim 10, wherein 32 parallel version are simulated using 32-bit panel simulation.

12. The method of claim 8, further comprising:

repeating steps m) through q) until an inversion fix point is reached ending the inversion detection phase, the inversion fix point occurring after completion of a number of repeated iterations during which no further polarity determinations are made.

13. The method of claim 12, further comprising:

r) grouping, in the functional phase, all unmapped state elements in an initial class;

s) inputting equivalent random values to each of the unmapped state elements; and

t) refining the unmapped state elements into equivalence classes based upon at least one of an equivalent output property and a precisely opposite output property generated in response to the input random values.

14. The method of claim 12, wherein random bit values are input to mapped inputs and a third value is input to unmapped inputs.

15. The method of claim 13, further comprising:

u) determining equivalence classes which include two state elements, one state element being from each of the first and second circuits;

v) mapping the two state elements in the determined equivalence classes to one another; and

w) updating the fanin and fanout partial signatures of the state elements of the first and second circuits based on the mapping between the two state elements.

16. The method of claim 15, further comprising:

repeating steps s) through w) until a functional fix point is reached, the functional fix point occurring after a completion of one of:

I) a number of repeated iterations during which no further mappings are determined; and

II) a mapping of all state elements of the first and second circuits; and

outputting mappings and equivalence classes.

17. The method of claim 1, further comprising:

comparing, in a second structural phase, structural features of state elements in the first circuit to structural features of state elements in the second circuit for equivalence;

determining, during the second structural phase, mappings between state elements of the first circuit and the second circuit based on the comparison of the structural features;

detecting, in a second inversion detection phase, the polarity of the mappings; and

outputting final mappings as a function of the detection;

wherein, in each phase, results from all previous phases are used to improve inputs.

18. An article comprising a computer-readable medium which stores computer-executable instructions for causing a computer system to:

compare, in a structural phase, structural features of state elements in the first circuit to structural features of state elements in the second circuit for equivalence;

determine, during the structural phase, mappings between state elements of the first circuit and the second circuit based on the comparison of the structural features;

account for don't care input conditions before comparing state element function;

detect, in an inversion detection phase, the polarity of the mappings;

compare, in a functional phase, the functionality of state elements in the first circuit to state elements in the second circuit for equivalence using a three-valued simulation;

determine further mappings based upon the functional comparison during the functional phase; and

detect whether a threshold condition for completion of the mapping process is satisfied.

19. The article of claim 18 which further stores instructions causing a computer system to:

input random values to each element of a pair of mapped state elements during the inversion detection phase, the random values being bit values for mapped inputs having a known polarity and being a third value for unmapped inputs and mapped inputs having an unknown polarity;

compare output values from each element of the pair of state elements based on the input random values; and

determine whether the pair is one of direct and inverse mapped based on the comparison.

20. A computer system for automatically mapping state elements between a first circuit and a second circuit, comprising:

an input interface; and

a processor, the processor configured to:

compare, in a structural phase, structural features of state elements in the first circuit to structural features of state elements in the second circuit for equivalence;

determine, during the structural phase, mappings between state elements of the first circuit and the second circuit based on the comparison of the structural features;

account for don't care input conditions specified using the input interface before comparing state element function;

detect, in an inversion detection phase, the polarity of the mappings;

compare, in a functional phase, the functionality of state elements in the first circuit to state elements in the second circuit for equivalence using a three-valued simulation;

determine further mappings based upon the functional comparison during the functional phase; and

detect whether a threshold condition for completion of the mapping process is satisfied.

21. The computer system of claim 20, wherein the processor is further configured to:

input random values to each element of a pair of mapped state elements during the inversion detection phase, the random values being bit values for mapped inputs having a known polarity and being a third value for unmapped inputs and mapped inputs having an unknown polarity;

compare output values from each element of the pair of state elements based on the input random values; and

determine whether the pair is one of direct and inverse mapped based on the comparison.